**Question 1**

**1. Direct mapped**

No.blocks 214 (blocks)

* Size of index = 14 bSits

Size of offset = log2(Size of block) = log2(256) = 8 bits

Size of index = Size of address – Size of index – Size of offset = 32 – 14 − 8 =10 bits

**2. 4-way set associative**

No.sets 212 (blocks)

* Size of index = 12 bits

Size of offset = log2(Size of block) = log2(256) = 8 bits

Size of tag = Size of address – Size of index – Size of offset =32 – 12 – 8 = 12bits

**3. Fully associative**

Size of index = 0

Size of offset = log2(Size of block) = log2(256) = 8 bits

Size of tag = Size of address – Size of index – Size of offset = 32 – 0 – 8 = 24bits

**Question 2**

index = 0 tag = 27 – 0 − 7 = 20

**1. Direct mapped**

Memory = 256 MB => Address bits =

Half-word access => Offset = log2(64.2) = 7

**2. 8-way set associative**

index = = 10

tag = 27 − 10 − 7 = 10

**3. Fully associative**

index = = 9

tag = 27 – 9 − 7 = 11

**Question 3**

**1. Direct mapped**

Number of block cache: = 1 = 24

* 4 block caches

|  |  |  |
| --- | --- | --- |
| Address | n mod 4 | Status |
| 0 | 0 | miss |
| 4 | 0 | hit |
| 1 | 1 | miss |

|  |  |  |
| --- | --- | --- |
| 5 | 1 | hit |
| 65 | 1 | hit |
| 1 | 1 | hit |
| 67 | 3 | miss |
| 46 | 2 | hit |
| 1 | 1 | hit |
| 70 | 2 | hit |
| 2 | 2 | hit |
| 0 | 0 | hit |

**2. 2-way set associative**

Number of block cache: = 1 = 22

* 2 block caches

|  |  |  |
| --- | --- | --- |
| Address | n mod 4 | Status |
| 0 | 0 | miss |
| 4 | 0 | hit |
| 1 | 1 | hit |
| 5 | 1 | hit |
| 65 | 1 | hit |
| 1 | 1 | hit |
| 67 | 1 | hit |
| 46 | 0 | hit |
| 1 | 1 | hit |
| 70 | 0 | hit |
| 2 | 0 | hit |
| 0 | 0 | hit |

**3. 4-way set associative**

Number of caches: = 1 = 20

* 1 block caches

|  |  |  |
| --- | --- | --- |
| Address | n mod 4 | Status |
| 0 | 0 | miss |
| 4 | 0 | hit |
| 1 | 0 | hit |
| 5 | 0 | hit |
| 65 | 0 | hit |
| 1 | 0 | hit |
| 67 | 0 | hit |
| 46 | 0 | hit |
| 1 | 0 | hit |
| 70 | 0 | hit |
| 2 | 0 | hit |
| 0 | 0 | hit |

**Question 4**

• Page: These cache blocks are called pages (size is P = 2p bytes).

• Page fault: reference to VM word that is not in physical memory (==cache miss).

• Cache miss: page fault.

• Write back/ Write through:

+ Write-back (or Write-behind): Writing is done only to the cache. A modified cache block is written back to the store, just before it is replaced.

+ Write-through: When data is updated, it is written to both the cache and the back-end storage

• PTE: Page table: an array of page table entries (PTEs) that maps virtual pages to physical pages. (== tags).

• TLB: Translation Lookaside Buffer : + Small hardware cache in MMU

+ Maps virtual page numbers to physical page numbers + Contains complete page table entries for small number of pages

**Question 5**

**Calculate the average CPI of a pipeline system where the miss rate of instruction memory is 5%, the miss rate of data memory is 10%. Miss penalty is 100 cycles. Base CPI is 1.5. The proportion of load/store instructions is 36%.**

Instruction miss cycles =xMiss rate x Miss penalty = I x 0.05 x 100

= 5 I (I is # of instructions)

Data miss cycles = x Miss rate x Miss penalty = I x 0.36 x 0.1 x 100 = 3.6 I

Total memory stall cycles = 5 I + 3.6 I = 8.6 I CPIstall = 1.5 + 8.6 = 10.1

= = = 6.733